H8/300L Series

Simultaneous Transmission/Reception in Synchronous Mode (H8/3644)

Introduction

Four bytes of 8-bit data are simultaneously transmitted and received using the serial data transfer function in synchronous mode. Data are transmitted and received LSB first.

Target Device

H8/3644

Contents

1. Specifications 2
2. Description of Functions 2
3. Principles of Operation 5
4. Description of Software 6
5. Flowchart 8
6. Program Listing 9

1. Specifications

1. As shown in figure 1.1, four bytes of eight-bit data are simultaneously transmitted and received using the serial data transfer function in synchronous mode.
2. Simultaneous transmission/reception is performed at a transfer clock cycle of 12.8 µs using an internal clock.
3. The length of transmit and receive data is eight bits and data is transmitted/received LSB first, which means the lowest bit of data is transmitted/received first.

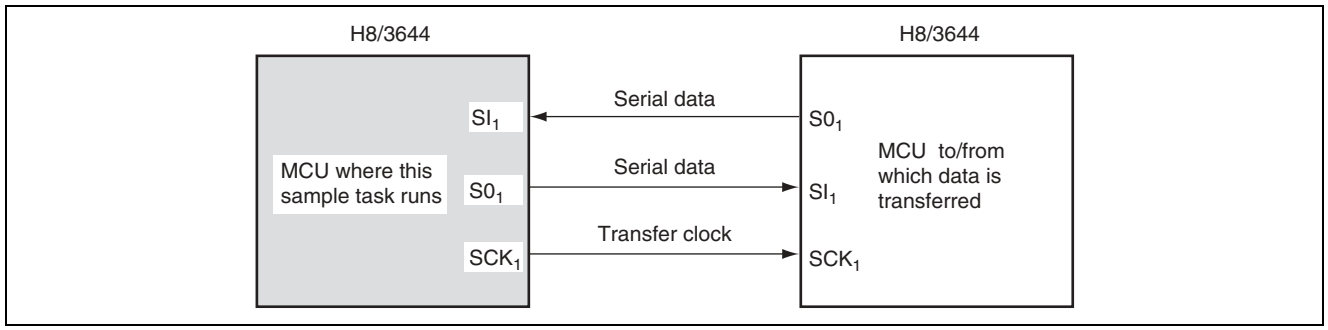


Figure 1.1 Simultaneous Serial Transmission/Reception in Synchronous Mode

2. Description of Functions

1. In this sample task, serial data is simultaneously transmitted/received in synchronous mode via the serial communication interface (SCI). Figure 2.1 shows a block diagram of simultaneous serial transmission/reception in synchronous mode, and the following is the description for the block diagram:
   - The frequency of the system clock (ϕ) used as the basic clock for the CPU or peripheral-function operation is 5-MHz; this clock is obtained by dividing the 10-MHz OSC clock by 2.
   - The prescaler S (PSS) is a 13-bit counter, to which ϕ is input. PSS counts up on each cycle.
   - The serial control register 1 (SCR1) is an 8-bit readable/writable register that selects operating mode, transfer clock source, and prescaler division ratio.
   - The serial control/status register 1 (SCSR1) is an 8-bit register that indicates operation status, error status, etc. If the synchronous clock is input continuously after the reception completion, reception is not performed but ORER in SCSR1 is set to 1 to indicate an overrun state.
   - The serial data register U (SDRU) is an 8-bit readable/writable register that functions as a data register for the upper 8 bits in 16-bit data transfer. Data written to SDRU is output to SDRL with the LSB first. Then, data is in turn input from the SI_1 pin with the LSB first, and data is shifted from the MSB to the LSB.
   - The serial data register L (SDRL) is an 8-bit readable/writable register that functions as a data register in 8-bit data transfer and as a data register for the lower 8 bits in 16-bit data transfer. In 8-bit data transfer, data written to SDRL is output from the SO_1 pin with the LSB first. Then, data is in turn input from the SI_1 pin with the LSB first, and data is shifted from the MSB to the LSB. In 16-bit data transfer, operation is the same as that in 8-bit data transfer except that data is input from SDRU.
   - SDRU and SDRL should be read or written to after data transmission or reception is complete. If they are read or written to during data transmission or reception, data may not be guaranteed.
   - The transfer clock can be selected from eight internal clocks and external clocks. When an internal clock is selected, the SCK_1 pin is used as an output pin. The selected clock is continuously output from the SCK_1 pin if clock continuous output mode is set. When an external clock is selected, the SCK_1 pin is used as the clock input pin.
   - In this sample task, settings are made so that the transfer clock source is PSS, the frequency of prescaler is divided by 64, and the transfer clock cycle is 12 µs.
   - The SCI1 transfer data format can be selected from 8 bits and 12 bits. Data is transferred in the LSB first method that transmits or receives data from the lowest bit. Transmit data is output from the falling edge of the transfer clock to the next rising edge. Receive data is acquired on the rising edge of the transfer clock.
   - In this sample task, the 8-bit operating mode is set to perform 8-bit data transmission/reception.

— The SCI1 clock pin (SCK_1) functions as a clock input/output pin for the SCI1.
— The SCI1 data output pin (SO_1) functions as a transmit data output pin for the SCI1.
— The SCI1 data input pin (SI_1) functions as a receive data input pin for the SCI1.
— When the SCI1 completes data transfer, the SCI1 interrupt request flag bit (IRRS1) in the interrupt request register 2 (IRR2) is set to 1. SCI1 interrupt requests can be enabled/disabled with the SCI1 interrupt enable bit (IENS1) in the interrupt enable register 2 (IENR2).

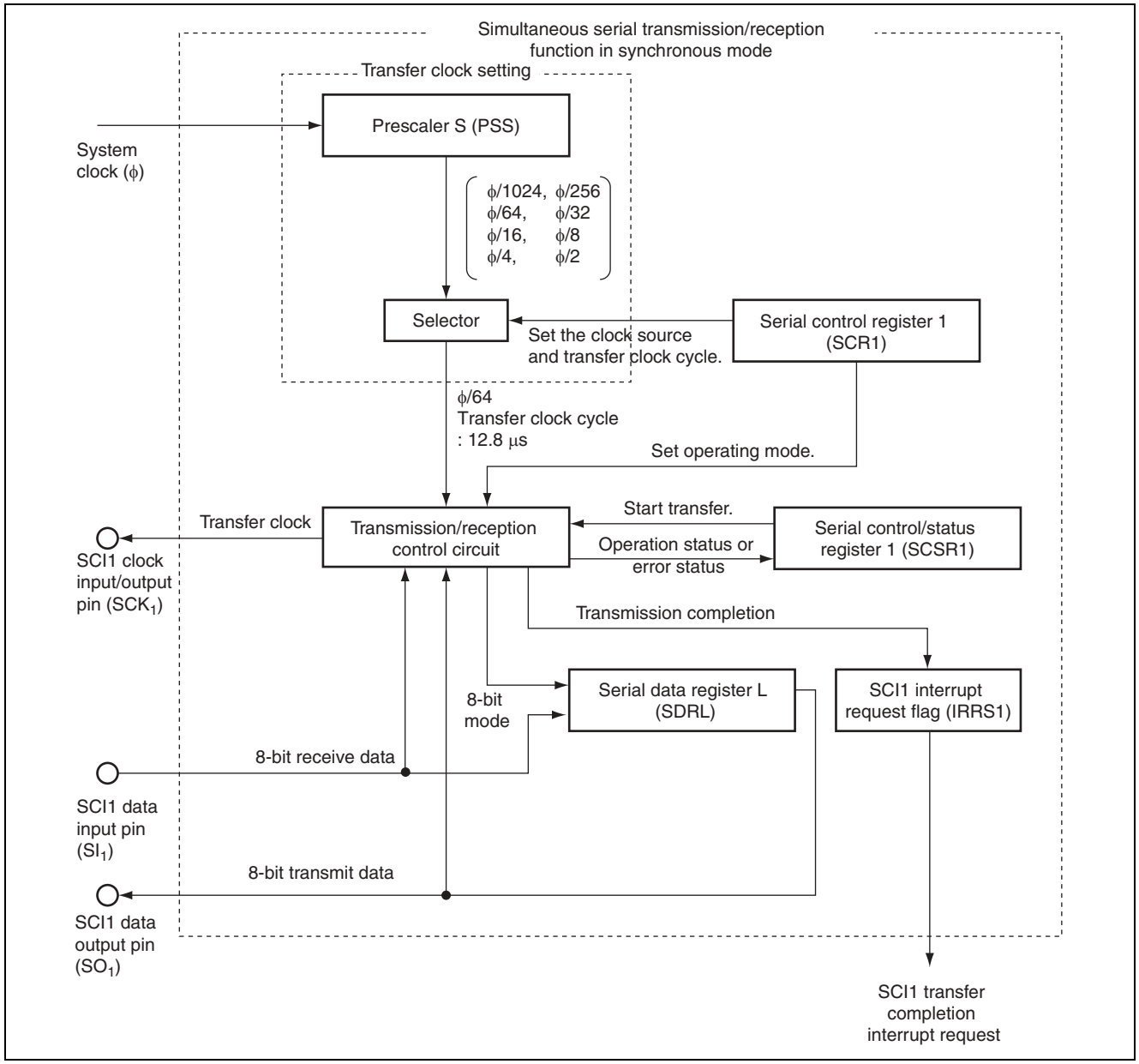


Figure 2.1 Block Diagram of Simultaneous Serial Transmission/Reception Function in Synchronous Mode

2. Table 2.1 shows the allocation of functions used in this sample task. Functions are allocated as shown in table 1 to perform simultaneous serial transmission/reception in synchronous mode.

Table 2.1 Function Allocation

Function	Function Allocation
SCR1	Operating mode, transfer clock source and prescaler division ratio are set.
SCSR1	Operation status or error status is indicated.
SDRL	Data register for 8-bit transmit/receive data
SCK_1	Transfer clock output pin of SCI1
SO_1	Transmit data output pin of SCI1
SI_1	Receive data input pin of SCI1
IRRS1	SCI1 transfer completion is indicated.
IENS1	Enabling/disabling of SCI1 interrupt requests is controlled.
PMR3	$P3_2/SO_1$, $P3_1/SI_1$ and $P3_0/SCK_1$ pin functions are set.
PMR7	Turning on/off of the $P3_2/SO_1$ pin output buffer PMOS is controlled.

3. Principles of Operation

1. Figure 3.1 shows the principles of operation. Simultaneous serial transmission/reception is performed in synchronous mode with the hardware and software processing shown in the figure.

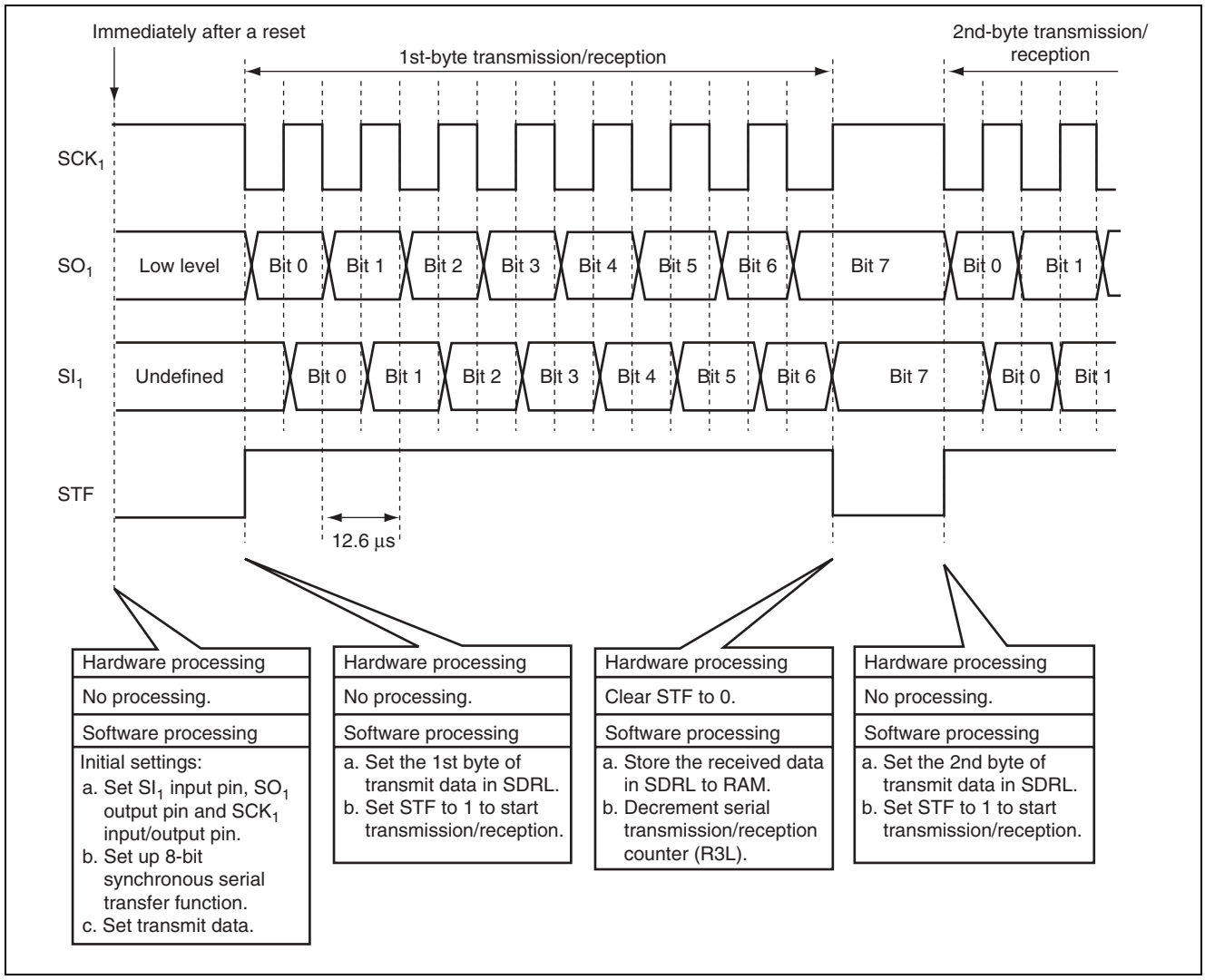


Figure 3.1 Operation Principles of Simultaneous Transmission/Reception in Synchronous Mode

4. Description of Software

4.1 Module

Table 4.1 describes the module used in this sample task.

Table 4.1 Description of Module

Module	Label	Function
Main routine	main	Initializes the stack pointer, sets transfer data, sets for synchronous serial data transmission/reception, enables interrupts, stores receive data to RAM, and ends when 4 bytes of data have been transmitted/received.

4.2 Arguments

Table 4.2 describes the arguments used in this sample task.

Table 4.2 Description of Modules

Argument	Function	Used in	Data Length	Input/ Output
STD0 to STD3	Data for synchronous serial transmission	Main routine	1 byte	Input
SRD0 to SRD3	Data received in synchronous mode	Main routine	1 byte	Output

4.3 Internal Registers

The internal registers used in this sample task are described in table 4.3.

Table 4.3 Description of Internal Registers

Register		Function	Address	Setting
SCR1	SNC1 SNC0	Serial Control Register 1 (Operating Mode Select 1, 0) When SNC1 = 0 and SNC0 = 0, operating mode is set to 8-bit mode.	H'FFA0 Bit 7 Bit 6	SNC1 = 0 SNC0 = 0
	MRKON	Serial Control Register 1 (Tail Mark Control) When MRKON = 0, a tail mark is not output.	H'FFA0 Bit 5	0
	CKS3	Serial Control Register 1 (Clock Source Select 3) When CKS3 = 0, prescaler S is selected for the clock source.	H'FFA0 Bit 3	0
	CKS2 CKS1 CKS0	Serial Control Register 1 (Clock Source Select 2, 1, 0) When CKS2 = 0, CKS1 = 1 and CKS0 = 0, prescaler division ratio is set to 64 and the transfer clock cycle is set to 12.8 μs.	H'FFA0 Bit 2 Bit 1 Bit 0	CKS2 = 0 CKS1 = 1 CKS0 = 0
SCSR1	SOL	Serial Control/Status Register 1 (Expansion Data Bit) When SOL = 0, the SO_1 pin output level is changed to low. When SOL = 1, the SO_1 pin output level is changed to high.	H'FFA1 Bit 6	0
	ORER	Serial Control/Status Register 1 (Overrun Error Flag) When ORER = 0, indicates that no overrun error occurred. When ORER = 1, indicates that an overrun error occurred.	H'FFA1 Bit 5	0
	STF	Serial Control/Status Register 1 (Start Flag) When STF = 0, transfer operation is complete. When STF = 1, transfer operation starts.	H'FFA1 Bit 0	0

Register		Function	Address	Setting
SDRL		Serial Data Register L Stores 8-bit receive data during 8-bit transfer	H'FFA3	—
IENR2	IENS1	Interrupt Enable Register 2 (SCI1 Interrupt Enable) When IENS1 = 0, SCI1 interrupt requests are disabled. When IENS1 = 1, SCI1 interrupt requests are enabled.	H'FFF5 Bit 4	0
IRR2	IRRS1	Interrupt Request Register 2 (SCI1 Interrupt Request Flag) When IRRS1 = 0, SCI1 interrupt requests are not requested. When IRRS1 = 1, SCI1 interrupt requests are requested.	H'FFF8 Bit 4	0
PMR3	SO1	Port Mode Register 3 ($P3_2$/SO_1 Pin Function Switch) When SO1 = 1, this pin functions as SO_1 output pin.	H'FFFD Bit 2	1
	SI1	Port Mode Register 3 ($P3_1$/SI_1 Pin Function Switch) When SI1 = 1, this pin functions as SI_1 output pin.	H'FFFD Bit 1	1
	SCK1	Port Mode Register 3 ($P3_0$/SCK_1 Pin Function Switch) When SCK1 = 1, this pin functions as SCK_1 input/output pin.	H'FFFD Bit 0	1
PMR7	PQF1	Port Mode Register 7 ($P3_2$/SO_1 Pin PMOS Control) When POF1 = 0, CMOS output is performed.	H'FFFF Bit 0	0

4.4 Description of RAM

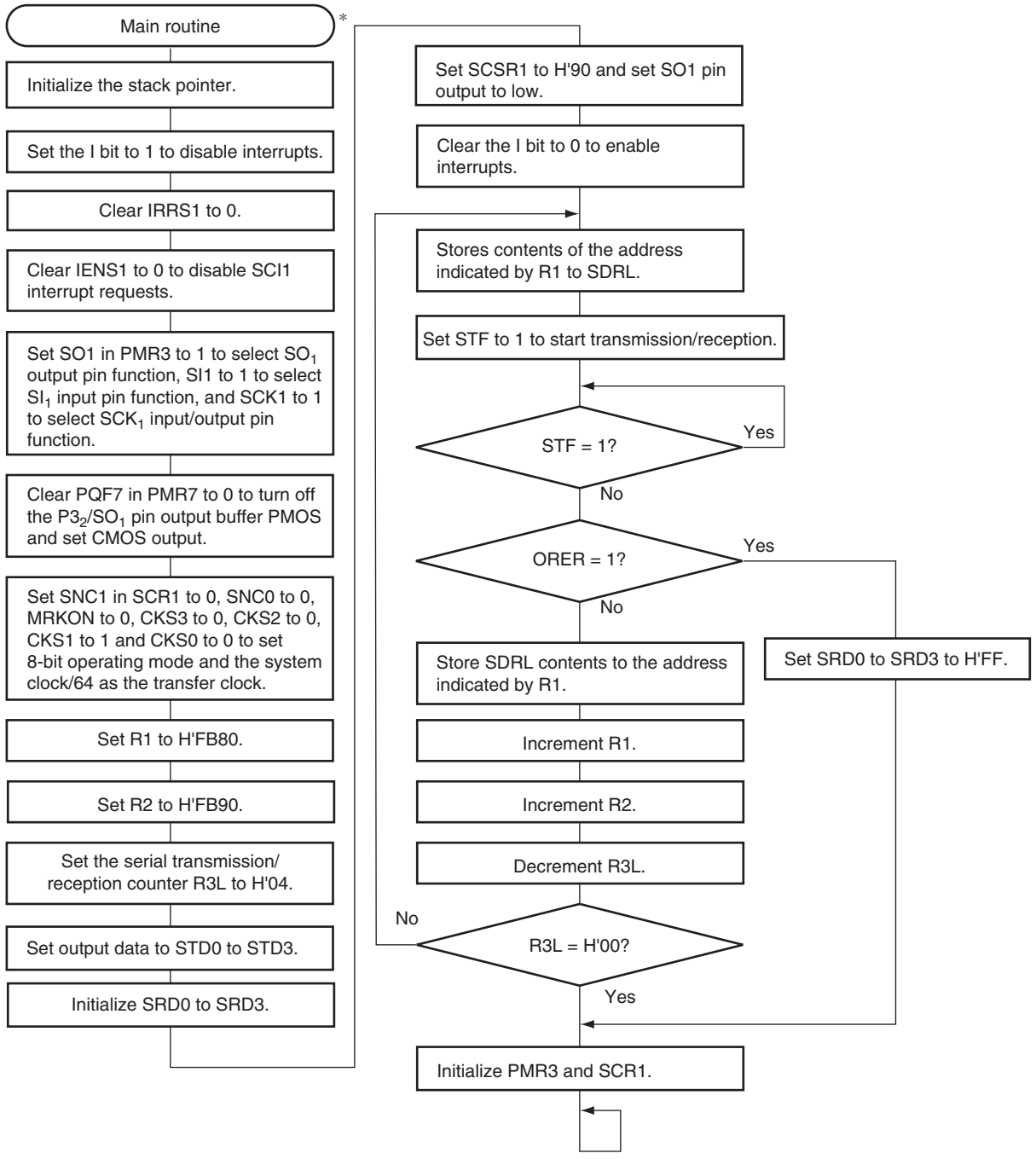
Table 4.4 describes the RAM used in this sample task.

Table 4.4 Description of RAM

Label	Function	Address	Used in
STD0	Stores the first byte of data for synchronous serial transmission.	H'FB80	Main routine
STD1	Stores the second byte of data for synchronous serial transmission.	H'FB81	Main routine
STD2	Stores the third byte of data for synchronous serial transmission.	H'FB82	Main routine
STD3	Stores the fourth byte of data for synchronous serial transmission.	H'FB83	Main routine
SRD0	Stores the first byte of data received in synchronous mode.	H'FB90	Main routine
SRD1	Stores the second byte of data received in synchronous mode.	H'FB91	Main routine
SRD2	Stores the third byte of data received in synchronous mode.	H'FB92	Main routine
SRD3	Stores the fourth byte of data received in synchronous mode.	H'FB93	Main routine

5. Flowchart

1. Main routine


Main routine
*
Initialize the stack pointer.
Set the I bit to 1 to disable interrupts.
Clear IRRS1 to 0.
Clear IENS1 to 0 to disable SCI1 interrupt requests.
Set SO1 in PMR3 to 1 to select SO1 output pin function, SI1 to 1 to select SI1 input pin function, and SCK1 to 1 to select SCK1 input/output pin function.
Clear PQF7 in PMR7 to 0 to turn off the P32/SO1 pin output buffer PMOS and set CMOS output.
Set SNC1 in SCR1 to 0, SNC0 to 0, MRKON to 0, CKS3 to 0, CKS2 to 0, CKS1 to 1 and CKS0 to 0 to set 8-bit operating mode and the system clock/64 as the transfer clock.
Set R1 to H'FB80.
Set R2 to H'FB90.
Set the serial transmission/ reception counter R3L to H'04.
Set output data to STD0 to STD3.
Initialize SRD0 to SRD3.
Set SCSR1 to H'90 and set SO1 pin output to low.
Clear the I bit to 0 to enable interrupts.
Stores contents of the address indicated by R1 to SDRL.
Set STF to 1 to start transmission/reception.
STF = 1?
Yes
No
ORER = 1?
Yes
No
Store SDRL contents to the address indicated by R1.
Set SRD0 to SRD3 to H'FF.
Increment R1.
Increment R2.
Decrement R3L.
No
R3L = H'00?
Yes
Initialize PMR3 and SCR1.

Note: * In this sample task, the stack pointer is set in INIT.SRC (assembly language).

6. Program Listing

```
;*************************************************************************************
;*                                                                                   *
;*     H8/300L Series -H8/3644,H8/3657-                                              *
;*     Application Note                                                              *
;*                                                                                   *
;*     'Synchronous Serial Data Simultaneous                                         *
;*      Transmission/Reception'                                                      *
;*                                                                                   *
;*     Function                                                                      *
;*     : Serial Communication Interface                                              *
;*       Synchronous Serial Interface                                                *
;*       -Transmitting/Receiving                                                     *
;*                                                                                   *
;*     External Clock : 10MHz                                                        *
;*     Internal Clock :  5MHz                                                        *
;*     Sub Clock      : 32.768kHz                                                    *
;*                                                                                   *
;*************************************************************************************
;
          .cpu         300L
;
;*************************************************************************************
;*  Symbol Defnition                                                                 *
;*************************************************************************************
;
SCR1      .equ         H'FFA0        ;Serial Control Register 1
SNC1      .bequ        7,SCR1        ;Select the Operation Mode 1
SNC0      .bequ        6,SCR1        ;Select the Operation Mode 0
MRKON     .bequ        5,SCR1        ;TAIL MARK Control
LTCH      .bequ        4,SCR1        ;LATCH TAIL Select
CKS3      .bequ        3,SCR1        ;Clock Source Select 3
CKS2      .bequ        2,SCR1        ;Clock Select 2
CKS1      .bequ        1,SCR1        ;Clock Select 1
CKS0      .bequ        0,SCR1        ;Clock Select 0
SCSR1     .equ         H'FFA1        ;Serial Control Status Register 1
SOL       .bequ        6,SCSR1       ;Extended Data Bit
ORER      .bequ        5,SCSR1       ;Overrun Error Flag
MTRF      .bequ        1,SCSR1       ;TAIL MARK Transmit Flag
STF       .bequ        0,SCSR1       ;Start Flag
SDRU      .equ         H'FFA2        ;Serial Data Register U
SDRL      .equ         H'FFA3        ;Serial Data Register L
IENR2     .equ         H'FFF5        ;Interrupt Enable Register 2
IENS1     .bequ        4,IENR2       ;SCI1 Interrupt Enable
IRR2      .equ         H'FFF8        ;Interrupt Request Register 2
IRRS1     .bequ        4,IRR2        ;SCI1 Interrupt Request Flag
PMR3      .equ         H'FFFD        ;Port Mode Register 3
SO1       .bequ        2,PMR3        ;P32/SO1 Pin Function Switch
SI1       .bequ        1,PMR3        ;P31/SI1 Pin Function Switch
SCK1      .bequ        0,PMR3        ;P30/SCK1 Pin Function Switch
PMR7      .equ         H'FFFF        ;Port Mode Register 7
POF1      .bequ        0,PMR7        ;P32/SO1 Pin Function Switch
;
```

```
;*****************************************************************************
;*  RAM Allocation                                                           *
;*****************************************************************************
;
STACK     .equ          H'FF80        ;Stack Pointer
STD0      .equ          H'FB80        ;Serial Transmitting Data 0
STD1      .equ          H'FB81        ;Serial Transmitting Data 1
STD2      .equ          H'FB82        ;Serial Transmitting Data 2
STD3      .equ          H'FB83        ;Serial Transmitting Data 3
SRD0      .equ          H'FB90        ;Serial Receiving Data 0
SRD1      .equ          H'FB91        ;Serial Receiving Data 1
SRD2      .equ          H'FB92        ;Serial Receiving Data 2
SRD3      .equ          H'FB93        ;Serial Receiving Data 3
;
;*****************************************************************************
;*  Vector Address                                                           *
;*****************************************************************************
;
          .org          H'0000
          .data.w       MAIN          ;Reset Interrupt
;
          .org          H'0008
          .data.w       MAIN          ;IRQ0 Interrupt
          .data.w       MAIN          ;IRQ1 Interrupt
          .data.w       MAIN          ;IRQ2 Interrupt
          .data.w       MAIN          ;IRQ3 Interrupt
          .data.w       MAIN          ;INT0 - INT7 Interrupt
;
          .org          H'0014
          .data.w       MAIN          ;Timer A Interrupt
          .data.w       MAIN          ;Timer B1 Interrupt
;
          .org          H'0020
          .data.w       MAIN          ;Timer X Interrupt
          .data.w       MAIN          ;Timer V Interrupt
;
          .org          H'0026
          .data.w       MAIN          ;SCI1 Interrupt
;
          .org          H'002A
          .data.w       MAIN          ;SCI3 Interrupt
          .data.w       MAIN          ;A/D Converter Interrupt
          .data.w       MAIN          ;SLEEP Instruction Executed Interrupt
;
```

```
;*************************************************************************************
;*  Main Program                                                                     *
;*************************************************************************************
;
          .org          H'1000
;
MAIN      .equ          $
          MOV.W         #STACK,SP      ;Initialize Stack Pointer
          ORC           #H'80,CCR      ;Interrupt Disable
;
          BCLR          IRRS1          ;Clear IRRS1
          BCLR          IENS1          ;SCI1 Interrupt Disable
;
          MOV.W         #H'07F8,R0
          MOV.B         R0H,@PMR3      ;Initialize SO1 & SI1 & CKS1 Pin Function
          MOV.B         R0L,@PMR7      ;Initialize SO1 Pin Function
;
          MOV.B         #H'02,R0L
          MOV.B         R0L,@SCR1      ;Initialize Synchronous Serial Transfer Function
;
          MOV.W         #H'FB80,R1     ;Initialize Serial Transmitting Data Address
          MOV.W         #H'FB90,R2     ;Initialize Serial Receiving Data Address
          MOV.B         #H'04,R3L      ;Initialize Serial Data Counter
;
          MOV.W         #H'0055,R0
          MOV.B         R0H,@STD0      ;Set Serial Transmitting Data 0
          MOV.B         R0L,@STD1      ;Set Serial Transmitting Data 1
          MOV.W         #H'AAFF,R0
          MOV.B         R0H,@STD2      ;Set Serial Transmitting Data 2
          MOV.B         R0L,@STD3      ;Set Serial Transmitting Data 3
;
          MOV.B         #H'00,R0L
          MOV.B         R0L,@SRD0      ;Initialize Serial Receiving Data 0
          MOV.B         R0L,@SRD1      ;Initialize Serial Receiving Data 1
          MOV.B         R0L,@SRD2      ;Initialize Serial Receiving Data 2
          MOV.B         R0L,@SRD3      ;Initialize Serial Receiving Data 3
;
          MOV.B         #H'9C,R0L
          MOV.B         R0L,@SCSR1     ;Initialize SO1 Pin Output Level
;
          ANDC          #H'7F,CCR      ;Interrupt Enable
;
MAIN1     .equ          $;
          MOV.B         @R1,R0L        ;Load Serial Transmitting Data
          MOV.B         R0L,@SDRL      ;Save Serial Transmitting Data
;
          BSET          STF            ;Start Serial Transmitting/Receiving
;
```

```
MAIN2     .equ        $
          BTST        STF          ;End Serial Transmitting/Receiving ?
          BNE         MAIN2        ;No.
;
          BTST        ORER         ;Overrun Error Flag = 1 ?
          BNE         MAIN3        ;Yes.
;
          MOV.B       @SDRL,R0L         ;Load
          MOV.B       R0L,@R2      ;Save
;
          ADDS        #1,R1        ;Increment Serial Transmitting Data Address
          ADDS        #1,R2        ;Increment Serial Receiving Data Address
          DEC          R3L         ;Decrement Serial Data Counter
          BNE         MAIN1        ;Serial Data Counter = H'00 ? No.
;
          BRA         MAIN4
;
MAIN3     .equ        $
          MOV.B       #H'FF,R0L
          MOV.B       R0L,@SRD0    ;Overrun Error
          MOV.B       R0L,@SRD1    ;Overrun Error
          MOV.B       R0L,@SRD2    ;Overrun Error
          MOV.B       R0L,@SRD3    ;Overrun Error
;
MAIN4     .equ        $
          MOV.B       #H'00,R0L
          MOV.B       R0L,@PMR3    ;Initialize SI1 & SCK1 Pin Function
          MOV.B       R0L,@SCR1    ;Initialize Synchronous Serial Transfer Function
;
MAIN9     .equ        $
          BRA         MAIN9
;
          .end
```

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Dec.19.03	—	First edition issued

Keep safety first in your circuit designs!

1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.